المحاضرة الثامنة

2013\11\26

تحدثت المحاضرة السابقة عن الوراثة التي تحدد علاقة الصوف ببعضها .

وتحدثت عن محددات الوصول التي تحدد مجال رؤية المتغيرات والطرق داخل وخارج الصفوف.

وسنتابع في هذه المحاضرة عن الوراثة وبعض المفاهيم.

تعليمة final:و توضع هذه التعليمة اما الصفوف والطرق والأعضاء البيانية

- امام الصفوف :وتفيد ان هذا الصف نهائي غير قابل للتعديل ولا يمكن الوراثة منه.

```
final class A{….}
```

- أمام الطريقة :أي ان هذه الطريقة أصبحت نهائية وغير قابلة للتعديل من قبل الصفوف الوارثة و لا يمكن التحميل الزائد عليها.

```
final void print(){…}
```

- اما م الأعضاء البيانية(المتحولات):أي ان هذا المتحول بمجرد اخذ قيمة أصبح نهائيا وتفيد في موضعين وقت الترجمة تخبر المترجم أن هذا المتحول لن تتغير قيمته ووقت التنفيذ ان هذه قيمة ابتدئية لا يمكن تغييرها

```
class A{
    final int x;
}
A a1=new A();
A a2=new A();
a1.x=5;
a2.x=3;
a1.x=6;//فقط واحدة مرة قيمة تأخذx لأن خاطئية الكتابة هذه
```

لا يوجد في جافا تعليمة لكتابة ثوابت لذلك نلجا للتالي

```
class A{
final static int a=5;
}
```

بالعودة الى الوراثة في لغة جافا الوراثة المتعددة غير ممكنة أي لا يمكن لصفة أن يرث من أكثر من صف .

Exe(1): اكتب صفا يمثل مضلع ثم بالاعتماد على الصف السابق اكتب صفا يمثل مثلث واخر يمثل مربع واكتب طريقة لحساب مساحة الشكلين.

```
public abstract class Shape
{
    public String name;
    abstract double area();
} //End of class Shape
class Triangle extends Chape {
double q,a,z;
Trangle(double e,double d,double c){
    q=e;
    a=d;
    z=c;
}//end of triangle constructor method
```

Note:
أي طريقة من النوع privateهي ضمنيا طريقة final لأنه لا يمكن التعديل على الطرق الخاصة أو التحميل الزائد عليها ويمكن وضع finalأمام الطرق من النوع privateلكن ذلك لن يضيف أي معنى عليها.

Note:يمكن وضع البرنامج السابق في ملف واحد بشرط ان يكون اسم الملف من اسم الصف العام أي المسبوق بpublic

E-mail: imran.shamia@gmail.com

```
    Triangle(int i, int i0, int i1) {
     //   throw new UnsupportedOperationException("Not yet
implemented");
        //the length of triangle side

        q=i;
        a=i0;
        z=i1;
    }//end of constructor method

    @Override
    double area() {
       // throw new UnsupportedOperationException("Not supported
yet.");
    double p=q+a+z;
    p=p/2;
    return(Math.pow(p*(p-q)*(p-a)*(p-z),0.5));
    }//end of area method
}//end of triangle class

class Square extends Chape {
double z;
Square(double z){// length of square side
    this.z=z;

}//end of constructor method
    @Override
    double area() {
      //  throw new UnsupportedOperationException("Not supported
yet.");
        return (z*z);
    }//end of area method

}//end of square class

class My-shape
{
    public static void main(String[] args) {
        Trangle t=new Trangle(3,4,5);
        t.name="triangle";
        System.out.println("area of the triangle="+t.area());
        Square a=new Square(5);
        a.name="my square";
        System.out.println("area of the squar="+a.area());

    }//enf of main method

}//end of class
```

Note:دالة حساب المساحة تم تعريفها في الصف الاساسي بشكل abstractلأنه لا يوجد قانون عام لمساحة الأشكال ونقوم بإعادة كتابة هذه الطريقة في كل صف يرث عن هذا الصف وتعريفها.

E-mail: imran.shamia@gmail.com

ونتيجة البرنامج السابق سوف تكون

```
area of the triangle=6.0
area of the square=25.0
```

$Exe(2)$:اكتب صفا بلغة جافا يقوم بما يلي:

جمع مصفوفتين مربعتين من البعدn×n

ضرب مصفوفتين مربعتين

البحث عن عنصر ما في مصفوفة

باستخدام الصف السابق اكتب برنامج يقوم بإدخال مصفوفتين من بعد n مدخل وتطبيق الدوال السابقة.

```
class My_matrix
{
    int[][] A;

    public My_matrix(int a) {
         A=new int [a][a];
    }
    //method of inserting matrix
    void read(){
        for(int i=0;i<A.length;i++){//insert the array
           for(int j=0;j<A[0].length;j++)
               A[i][j]=Stdin.readInt();
           System.out.println();
        }//end of loop
    }//end of read method
    //printing mathod
    void print(int[][] A){
        for(int i=0;i<A.length;i++){//printing the transformed
matrex
           for(int j=0;j<A[0].length;j++)
               System.out.print(A[i][j]+"  ");
           System.out.println();
        }//end of loop
    }//end of print method
    //method of adding matrix
    int[][] sum2matrix(int[][] B){
        int[][] sum=new int[A.length][A.length];
        for(int i=0;i<A.length;i++)
            for(int j=0;j<A[0].length;j++)
                sum[i][j]=A[i][j]+B[i][j];
        return sum;
    }//end of adding method
```


E-mail: imran.shamia@gmail.com

```
    //method of moultiplay matrix
    int[][] mult2matrix(int[][] B){
        int[][] mult =new int[A.length][A.length];
        for(int i=0;i<A.length;i++)
            for(int j=0;j<A[0].length;j++){
                mult[i][j]=0;
                for(int k=0;k<A[0].length;k++)
                    mult[i][j]+=A[i][j]*B[k][j];

    }//end of loop
        return mult;

}//end of mult method

    boolean search(int z)
    {
        boolean b = false;
        for (int i = 0; i < A.length; i++)
            for (int j = 0; j < A[0].length; j++)
                if (A[i][j] == z)
                {
                    b = true;
                    return b;
                }
        return b;
    }
}
public class Main
{


    public static void main(String[] args) {

       int n;
       do{n=Stdin.readInt();
       }while(n<2);
       My_matrix m=new My_matrix(n);
       m.read();
       My_matrix w=new My_matrix(n);
       w.read();
       m.print(w.sum2matrix(m.A));//لسهولة تعريفها عند الطباعة لدالة وسيط وضعنا
فورا المجموع طباعة
       int x=Stdin.readInt();
       boolean t1=m.search(x);
       boolean t2=w.search(x);
       System.out.print(t1);
       System.out.print(t2);
    }
}
```


E-mail: imran.shamia@gmail.com